Московский государственный университет имени М.В.Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра Методов математического прогнозирования

Обзор пакета системы R

“ROCR”

Выполнила: студентка 317 группы

Лобачева Екатерина

Москва

2012

1. Краткое описание пакета

ROCR представляет собой пакет системы R для оценки и визуализации качества классификации на два класса. Есть возможность подсчета различных мер качества классификации и построения 2D графиков для отдельных мер или для зависимости одной меры от другой. Пакет очень прост в использовании, так как содержит всего три интуитивно понятные функции и два класса для хранения промежуточных данных.

2. Начало работы

Для работы с данным пакетом необходимо сделать следующее:

- Установить систему R
- Скачать следующие пакеты: bitops, gtools, gdata, caTools, KernSmooth, gplots, ROCR
- Установить вышеперечисленные пакеты (вкладка системы R *Пакеты -> Установить пакет(ы) из локальных zip-файлов…*)
- Включить вышеперечисленные пакеты в указанном порядке (вкладка системы R *Пакеты -> Включить пакет…*)

3. Подробное описание работы пакета

3.1. Входные данные

Для использования средств данного пакета необходимы данные об истинных (labels) и предсказанных каким-либо образом (predictions) метках классов выборки объектов.

При этом данные могут представлять собой описание как одной выборки, так и нескольких, например нескольких подвыборок при использовании кросс-валидации. Наборы истинных и предсказанных меток представляются в виде векторов или списков одинаковой длины. В случае нескольких выборок данные представляются в виде матрицы или фрейма, в котором каждый столбец представляет собой отдельную выборку – выборки могут быть только одинаковой длины, или в виде списка, каждый элемент которого описывает одну из выборок, в этом случае выборки могут иметь различную длину.

Истинные метки могут принимать только два значения, на которых должно быть задано отношение сравнения (заданы по умолчанию: 0<1 – да и для любых чисел, ‘a’<’b’, FALSE<TRUE, чтобы использовать свои метки, нужно задавать на них отношение, как это делать будет описано ниже). Если предсказанные и истинные метки являются числами, то предсказанные метки могут принимать сколько угодно различных значений (например, если истинные 0 и 1, то предсказанные вполне могут быть заданы промежуточными значениями между 0 и 1), если же предсказанные или истинные метки не являются числами, то предсказанные метки могут принимать только 2 значения, причем те же, что и истинные.

Класс, для которого метка больше, будем называть положительным, а другой – отрицательным.

3.2. Меры качества

В данном пункте опишем доступные в пакете меры качества.

Для начала введем некоторые обозначения:

- Y/Y^ – истинные/предсказанные некоторым образом метки класса для некоторой выборки объектов.
- +/- – положительный/отрицательный классы.
- P/N – количество объектов, отнесенных к положительному/отрицательному классу.
- TP/TN – количество объектов, правильно отнесенных к положительному/ отрицательному классу.
- FP/FN – количество объектов, ошибочно отнесенных к положительному/ отрицательному классу.

Теперь рассмотрим сами меры:

acc	Accuracy. $\mathrm{P}(\mathrm{Y}\hat{} = \mathrm{Y})$. Подсчет: $\frac{TP+TN}{N+P}$.
err	Error rate. $\mathrm{P}(\mathrm{Y}\hat{} <> \mathrm{Y})$. Подсчет: $\frac{FP+FN}{N+P}$.
fpr, fall	False positive rate, fallout. $\mathrm{P}(\mathrm{Y}\hat{} = +\mid \mathrm{Y} = -)$. Подсчет: $\frac{FP}{N}$.
tpr, rec, sens	True positive rate, recall, sensitivity. $\mathrm{P}(\mathrm{Y}\hat{} = +\mid \mathrm{Y} = +)$. Подсчет: $\frac{TP}{P}$.
fnr, miss	False negative rate, miss. $\mathrm{P}(\mathrm{Y}\hat{} = -\mid \mathrm{Y} = +)$. Подсчет: $\frac{FN}{P}$.
tnr, spec	True negative rate, specificity. $\mathrm{P}(\mathrm{Y}\hat{} = -\mid \mathrm{Y} = -)$. Подсчет: $\frac{TN}{N}$.
ppv, prec	Positive predictive value, precision. $\mathrm{P}(\mathrm{Y}\hat{} = + \mid \mathrm{Y} = +)$. Подсчет: $\frac{TP}{TP+FP}$.
npv	Negative predictive value. $\mathrm{P}(\mathrm{Y}\hat{} = - \mid \mathrm{Y} = -)$. Подсчет: $\frac{TN}{TN+FN}$.
pcfall	Prediction-conditioned fallout. $\mathrm{P}(\mathrm{Y} = -\mid \mathrm{Y}\hat{} = +)$. Подсчет: $\frac{FP}{TP+FP}$.
pcmiss	Prediction-conditioned miss. $\mathrm{P}(\mathrm{Y} = +\mid \mathrm{Y}\hat{} = -)$. Подсчет: $\frac{FN}{TN+FN}$.
rpp	Rate of positive predictions. $\mathrm{P}(\mathrm{Y}\hat{} = +)$. Подсчет: $\frac{TP+FP}{TP+FP+TN+FN}$.
rnp	Rate of negative predictions. $\mathrm{P}(\mathrm{Y}\hat{} = -)$. Подсчет: $\frac{TN+FN}{TP+FP+TN+FN}$.
phi, mat	Phi correlation coefficient, matthews correlation coefficient. Подсчет: $\frac{TP*TN-FP*FN}{\sqrt{(TP+FN)*(TN+FP)*(TP+FP)*(TN+FN)}}$. Дает число от -1 до 1, где 1 - идеальный прогноз, 0 указывает на случайный прогноз. Величины ниже 0 указывают на прогноз хуже случайного.

mi	Mutual information. I(Y^, Y) := H(Y) − H(Y |Y^), где H – энтропия.
chisq	Chi square test statistic. Сложная функция, описание по команде: ?chisq.test.
odds	Odds ratio. Подсчет:$\frac{TP*TN}{FN*FP}$.
lift	Lift value. Подсчет:$\frac{\mathrm{P}(\mathrm{Y}\hat{}=+\|\mathrm{Y}=+)}{P(\mathrm{Y}\hat{}=+)}$.
f	Precision-recall F measure. Выражается через precision (P), recall (R) и некоторый задаваемый коэффициент α из отрезка [0,1]. Подсчет: $\frac{1}{\alpha\frac{1}{P}+(1-\alpha)\frac{1}{R}}$**.** Значение α по умолчанию 0.5.
rch	ROC convex hull. ROC кривые (tpr vs fpr) с удаленными вогнутыми кривыми (полученными при неоптимальных значениях отсечки). Не может использоваться совместно с другими мерами.
auc	Area under the ROC curve. Не может использоваться совместно с другими мерами. Возможен подсчет частичной площади до определенного значения fpr, для этого нужно подавать параметр fpr.stop из отрезка [0,1].
prbe	Precision-recall break-even point. Отсечки, в которых precision и recall равны. Не может использоваться совместно с другими мерами.
cal	Calibration error. Абсолютная разница между точностью предсказания меток и результатом классификации. Эта мера вычисляется для всех значений отсечек путем проходом окном по всему пространству отсечек. Размер окна по умолчанию 100, его можно задать с помощью дополнительного параметра window.size. Используется только с вероятностными предсказаниями (от 0 до 1).
mxe	Mean cross-entropy. Используется только с вероятностными предсказаниями (от 0 до 1). Подсчет:$-\frac{1}{P+N}\left(\sum_{y_i=+}\ln(\hat{y}_i)+\sum_{y_i=-}\ln(1-\hat{y}_i)\right)$. Не может использоваться совместно с другими мерами.
rmse	Root-mean-squared error. Подсчет: $\sqrt{\frac{1}{P+N}\Sigma_i(y_i-\widehat{y_i})^2}$. Используется только с числовыми метками классов. Не может использоваться совместно с другими мерами.
sar	Комбинация нескольких мер для получения более устойчивой меры. Подсчет: 1/3 * (Accuracy + Area under the ROC curve + Root mean-squared error).
ecost	Expected cost. Имеет обязательную ось x – 'probability-cost function'. Не может использоваться совместно с другими мерами. Подробности: Drummond&Holte 2000,2004.
cost	Стоимость классификатора при явно заданных ценах за ошибку на положительном и отрицательном классах. Доступны дополнительные параметры cost.fp и cost.fn, которые и задают цену ошибочной классификации в положительный и отрицательный класс соответственно. По умолчанию оба параметра равны 1.

Часто используемые вместе пары мер:

- ROC curves: tpr vs fpr,
- Precision/recall graphs: prec vs rec,
- Sensitivity/specificity plots: sens vs spec,
- Lift charts: lift vs rpp.

3.3. Классы

Пакет содержит два класса: prediction и performance.

3.4.1. prediction class

Объекты этого класса предназначены для внутреннего представления исходных данных: истинных и предсказанных каким-то образом меток классов.

Компоненты:

Все списки могут иметь несколько элементов, если исходные данные состоят из более, чем одной выборки.

predictions	Список, каждый элемент которого представляет собой вектор предсказанных меток.
labels	Список, каждый элемент которого представляет собой вектор истинных меток.
cutoffs	Список, каждый элемент которого представляет собой вектор всех отсечек – всех возможных предсказанных меток (при этом добавляется значение Inf, метки сортируются в порядке убывания и удаляются повторы).
fp	Список, каждый элемент которого представляет собой вектор, который состоит из количеств неправильно классифицированных объектов положительного класса при разделении объектов на основе предсказанных меток на классы по отсечкам из соответствующего вектора cutoffs.
tp	То же, что fp, но для правильно классифицированных объектов положительного класса.
tn	То же, что fp, но для правильно классифицированных объектов отрицательного класса.
fn	То же, что fp, но для неправильно классифицированных объектов отрицательного класса.
n.pos	Список, каждый элемент которого содержит число объектов положительного класса при истинных метках.
n.neg	То же, что n.pos, но для объектов отрицательного класса.
n.pos.pred	Список, каждый элемент которого представляет собой вектор, который состоит из количеств объектов, отнесенных к положительному классу при разделении объектов на основе предсказанных меток на классы по отсечкам из соответствующего вектора cutoffs.

n.neg.pred	То же, что n.pos.pred, но для отрицательного класса.

3.4.2. performance class

Объекты этого класса предназначены для хранения результатов оценки качества классификации в форме предназначенной для построения графика (отдельно рассматриваются меры качества для осей и параметризация).

Компоненты:

Все списки могут иметь несколько элементов, если исходные данные состоят из более, чем одной выборки.

x.name	Название меры качества, используемой для оси x.
y.name	Название меры качества, используемой для оси y.
alpha.name	Название элемента, используемого для создания параметризованной кривой. Обычно это “none” или “cutoff”.
x.values	Список, каждый элемент которого представляет собой вектор, который состоит из значений меры качества *x* в точках соответствующего вектора alpha.values.
y.values	То же, что x.values, но для меры качества *y*.
alpha.values	Список, каждый элемент которого представляет собой вектор, который состоит из значений заданного параметра кривой.

Объект данного класса может иметь 4 разных вида (для каждого вида приведен пример его создания с помощью функции performance, описание которой будет ниже):

- Описывается поведение меры, зависящей от отсечки, на всех возможных значениях отсечки. Тогда в x записываются значения отсечки, а в y – этой меры, alpha остается пустой.
  Пример: performance(pred, “acc”)
- Описывается зависимость двух мер, зависящих от отсечки, друг от друга на всех возможных значениях отсечки. Тогда в x и y записываются значения мер, а в alpha - отсечки.
  Пример: performance(pred, “tpr”, “fpr”)
- Описывается поведение меры, для которой изначально заложена некоторая вторая ось. Тогда в x записываются значения по этой второй оси, а в y – этой меры, alpha остается пустой.
  Пример: performance(pred, “ecost”)
- Описывается поведение меры, являющейся скаляром. Тогда в *y* записываются значения меры, а *x* и alpha остаются пустыми.
  Пример: performance(pred, “auc”)

3.4. Функции

Пакет содержит три функции: prediction, performance и plot.

3.4.1. prediction

Функция для создания объекта класса prediction из исходных данных.

Вызов:

prediction(predictions, labels, label.ordering = NULL)

Аргументы:

predictions	Вектор, матрица, список или фрейм, содержащий предсказанные метки выборки объектов.
labels	Вектор, матрица, список или фрейм, содержащий истинные метки выборки объектов.
label.ordering	Отношение сравнения между метками класса по умолчанию можно изменить, поставив в аргумент вектор, содержащий метки отрицательного и положительного класса.

3.4.2. performance

Функция для создания объекта класса performance из объекта класса prediction.

Вызов:

performance(prediction.obj, measure, x.measure="cutoff", ...)

Аргументы:

prediction.obj	Объект класса prediction.
measure	Мера качества, используемая для оси y.
x.measure	Мера качества, используемая для оси x.
…	Дополнительные аргументы, которые определены для некоторых мер.

3.4.3. plot

Функция для визуализации объекта класса performance.

Вызов:

plot(x, y, ..., avg="none", spread.estimate="none", spread.scale=1, show.spread.at=c(), colorize=FALSE, colorize.palette=rev(rainbow(256,start=0, end=4/6)), colorkey=colorize, colorkey.relwidth=0.25, colorkey.pos="right", print.cutoffs.at=c(), cutoff.label.function=function(x) { round(x,2) }, downsampling=0, add=FALSE)

Аргументы:

x	Объект класса performance.
y	Не используется.
…	Дополнительные графические параметры для настройки различных компонент графика. Для обращения к параметру некоторой компоненты нужно пользоваться следующей записью: component.parameter. Доступны следующие компоненты: xaxis, yaxis, coloraxis, box, points, text, plotCI (погрешности), boxplot.При настройке параметров самого холста и кривых префикс указывать не нужно.
avg	Объект, описывающий отображение нескольких кривых (например, если данные содержат несколько выборок объектов, полученных при кросс-валидации, то и кривых получается несколько). Кривые можно усреднять различными способами: • none – кривые рисуются отдельно без усреднения, • horizontal – горизонтальное усреднение, • vertical – вертикальное усреднение, • threshold - усреднение по отсечкам.
spread.estimate	При включенном усреднении кривых, отклонение от средней кривой может быть визуализировано как: • stderror – окно стандартной ошибки, • stddev – окно стандартного отклонения, • boxplot – окно разброса.
spread.scale	Константа, на которую домножаются длины окон stderror и stddev.
show.spread.at	При вертикальном усреднении этот вектор задает позиции x, в которых производится визуализация. По умолчанию она производится в 11 равномерно распределенных по всему пространству значений x точках.
colorize	Логическое значение, показывающее, должна ли кривая быть раскрашена в соответствии с отсечками.
colorize.palette	Если colorize включено, то определяет цветовую палитру, в которой отображается диапазон отсечек.
colorkey	Логическое значение. Если TRUE, то в 4% граничной зоне рисуется цветовой ключ, показывающий отображение отсечек в цветовую палитру.
colorkey.relwidth	Константа от 0 до 1, определяющая часть 4% граничной зоны, которая отводится под цветовой ключ.
colorkey.pos	Определяет, где как рисуется цветовой ключ: вертикально справо или горизонтально сверху.
print.cutoffs.at	Вектор значений отсечек, которые нужно напечатать вдоль кривой в соответствующих точках.
cutoff.label.function	По умолчанию значения отсечек, выводимые на кривой и цветовом ключе, округляются до двух знаков после запятой. Используя этот параметр, можно задать некоторое преобразование отсечек перед выводом (например, округление или взятие логарифма).
downsampling	При очень больших размерах выборки, построение графиков может быть медленным, а их размеры слишком большими. В таких случаях можно строить графики только по части выборки. Данный параметр задает константу от 0 до 1, которая показывает, по какой части

	объектов нужно строить графики. Если значение больше 1, то графики строятся по всей выборке.
add	Если TRUE, то кривые добавляются к уже существующему графику, иначе создается новый график.

4. Примеры

Рассмотрим несколько примеров, показывающих работу пакета ROCR. Красным шрифтом будем выделять команды, подающиеся R, а синим - полученные выводы.

4.1. Структуры данных и классов

Простейший пример для понимания структуры данных.

```
> a=c(0.2,0.8,0,0.9)
> b=c(0,0,1,1)
> c=c(0.3,0.2)
> d=c(0,0)
> pr=list(a,c)
> l=list(b,d)
> pred <- prediction(pr, l)
> pred
```

```
An object of class "prediction"
Slot "predictions":
[[1]]
[1] 0.2 0.8 0.0 0.9

[[2]]
[1] 0.3 0.2


Slot "labels":
[[1]]
[1] 0 0 1 1
Levels: 0 < 1

[[2]]
[1] 0 0
Levels: 0 < 1


Slot "cutoffs":
[[1]]
[1] Inf 0.9 0.8 0.2 0.0

[[2]]
[1] Inf 0.3 0.2


Slot "fp":
[[1]]
[1] 0 0 1 2 2

[[2]]
[1] 0 1 2


Slot "tp":
[[1]]
[1] 0 1 1 1 2

[[2]]
[1] 0 0 0
```

```
Slot "tn":
[[1]]
[1] 2 2 1 0 0

[[2]]
[1] 2 1 0


Slot "fn":
[[1]]
[1] 2 1 1 1 0

[[2]]
[1] 0 0 0


Slot "n.pos":
[[1]]
[1] 2

[[2]]
[1] 0
```

```
Slot "n.neg":
[[1]]
[1] 2

[[2]]
[1] 2


Slot "n.pos.pred":
[[1]]
[1] 0 1 2 3 4

[[2]]
[1] 0 1 2


Slot "n.neg.pred":
[[1]]
[1] 4 3 2 1 0

[[2]]
[1] 2 1 0
```

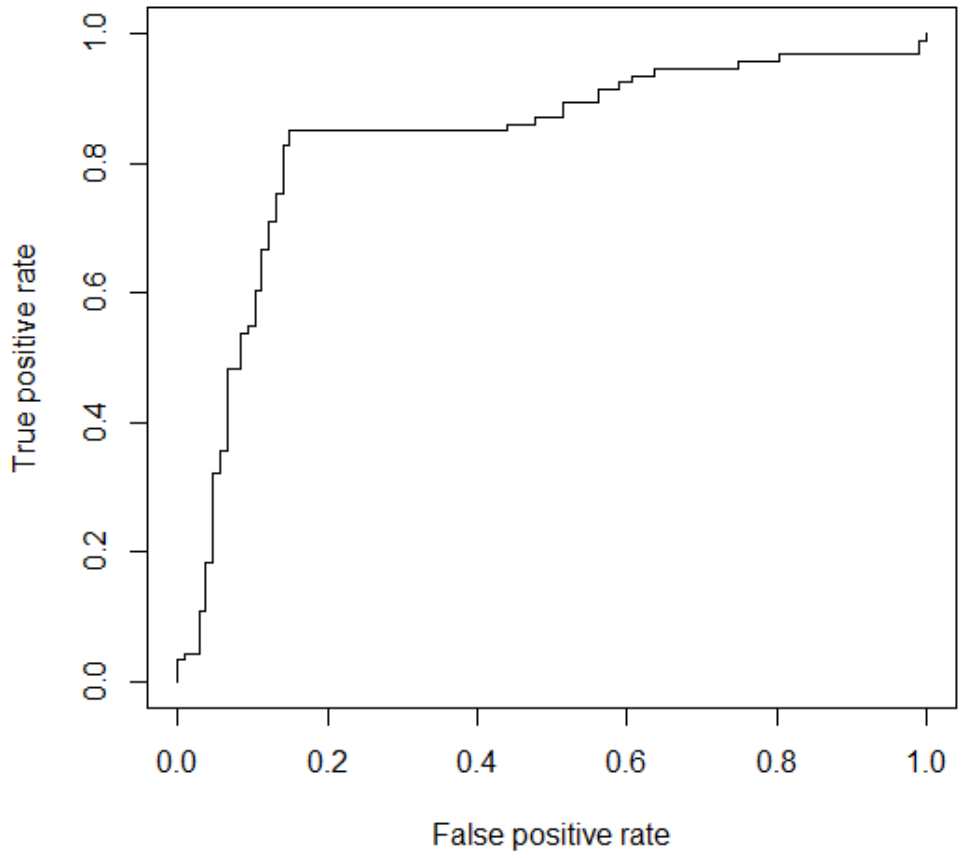
```
> perf1 <- performance(pred,"acc")
> perf1
```

```
An object of class "performance"
Slot "x.name":
[1] "Cutoff"

Slot "y.name":
[1] "Accuracy"

Slot "alpha.name":
[1] "none"

Slot "x.values":
[[1]]
[1] Inf 0.9 0.8 0.2 0.0

[[2]]
[1] Inf 0.3 0.2

Slot "y.values":
[[1]]
[1] 0.50 0.75 0.50 0.25 0.50

[[2]]
[1] 1.0 0.5 0.0

Slot "alpha.values":
list()
```

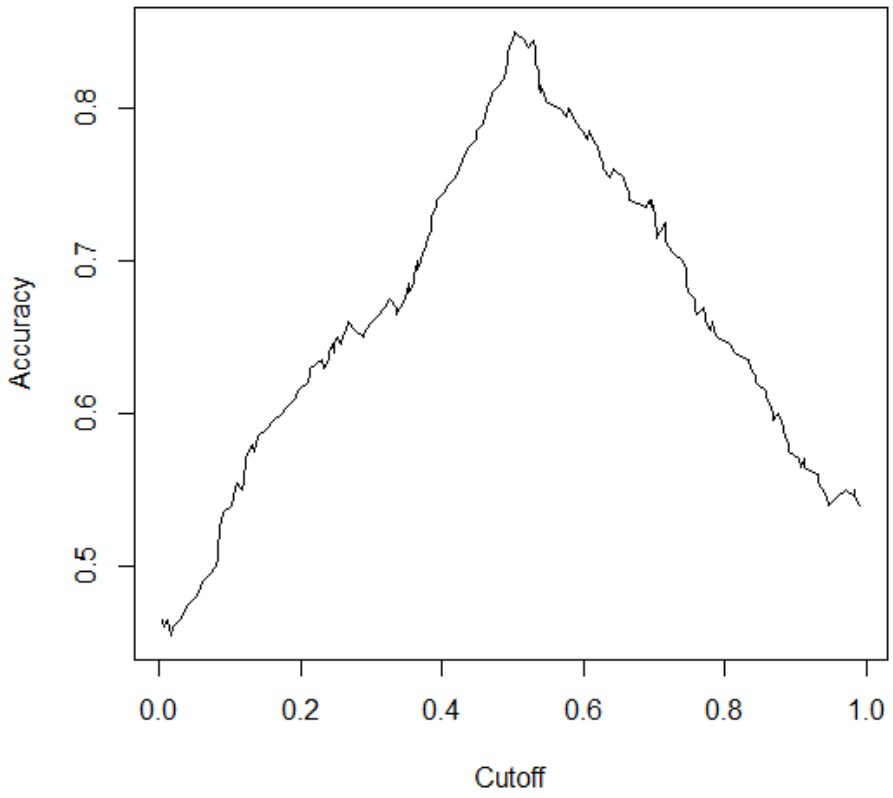
4.2. Пример с одной выборкой

Покажем различные возможности пакета на примере данных ROCR.simple.

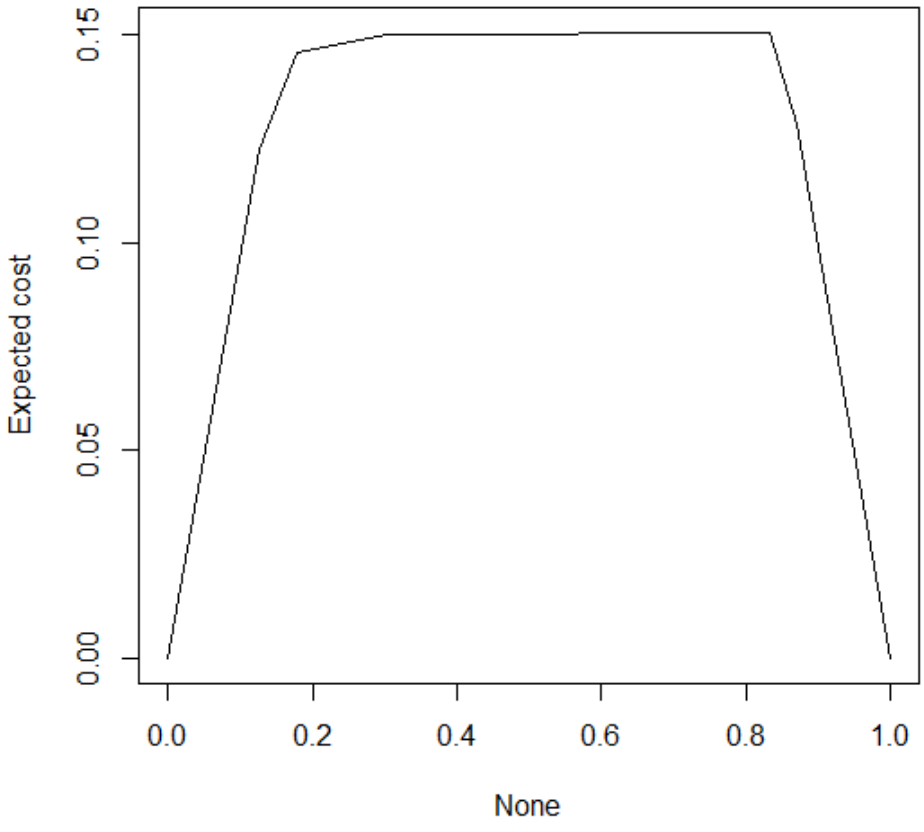
```
> data(ROCR.simple)
> pred <- prediction( ROCR.simple$predictions, ROCR.simple$labels)
> perf <- performance(pred,"tpr","fpr")
> plot(perf)
```



```
> perf <- performance(pred,"acc")
> plot(perf)
```



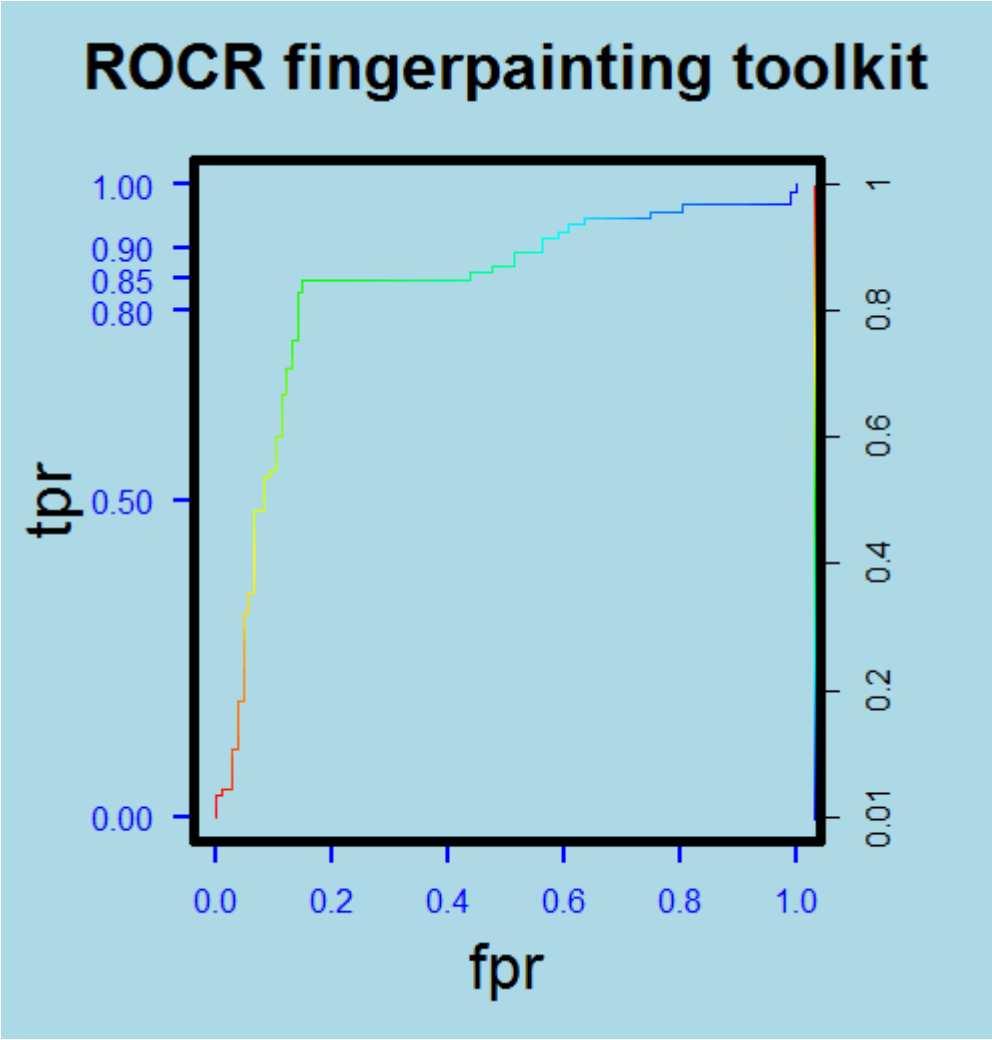

```
> perf <- performance(pred,"ecost")
> plot(perf)
```



```
> perf <- performance(pred,"auc")
```

```
An object of class "performance"
Slot "x.name":
[1] "None"

Slot "y.name":
[1] "Area under the ROC curve"

Slot "alpha.name":
[1] "none"

Slot "x.values":
list()

Slot "y.values":
[[1]]
[1] 0.8341875

Slot "alpha.values":
list()
```

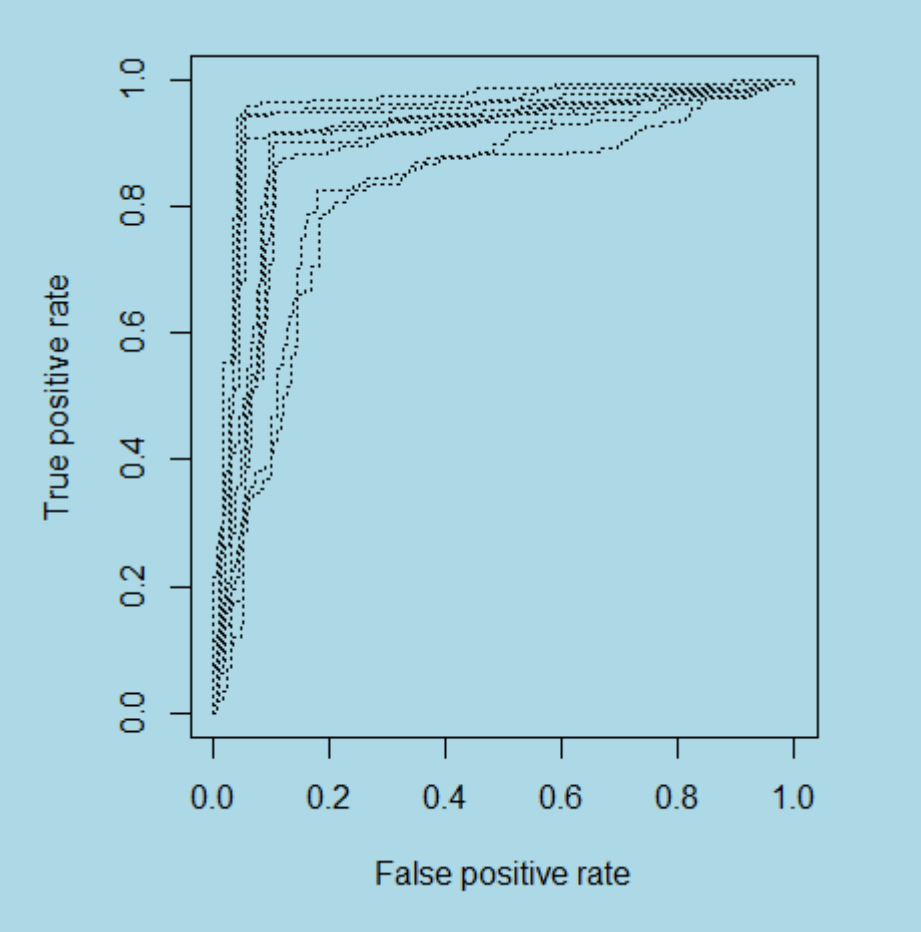
```
> perf <- performance(pred,"tpr","fpr")
> par(bg="lightblue", mai=c(1.2,1.5,1,1))
> plot(perf, main="ROCR fingerpainting toolkit", colorize=TRUE, xlab="fpr", ylab="tpr",
+box.lty=7, box.lwd=5, xaxis.col="blue", xaxis.col.axis="blue", yaxis.col="blue",
+yaxis.at=c(0,0.5,0.8,0.85,0.9,1), yaxis.las=1, xaxis.lwd=2, yaxis.lwd=2, yaxis.col.axis="blue",
+cex.lab=2, cex.main=2)
```



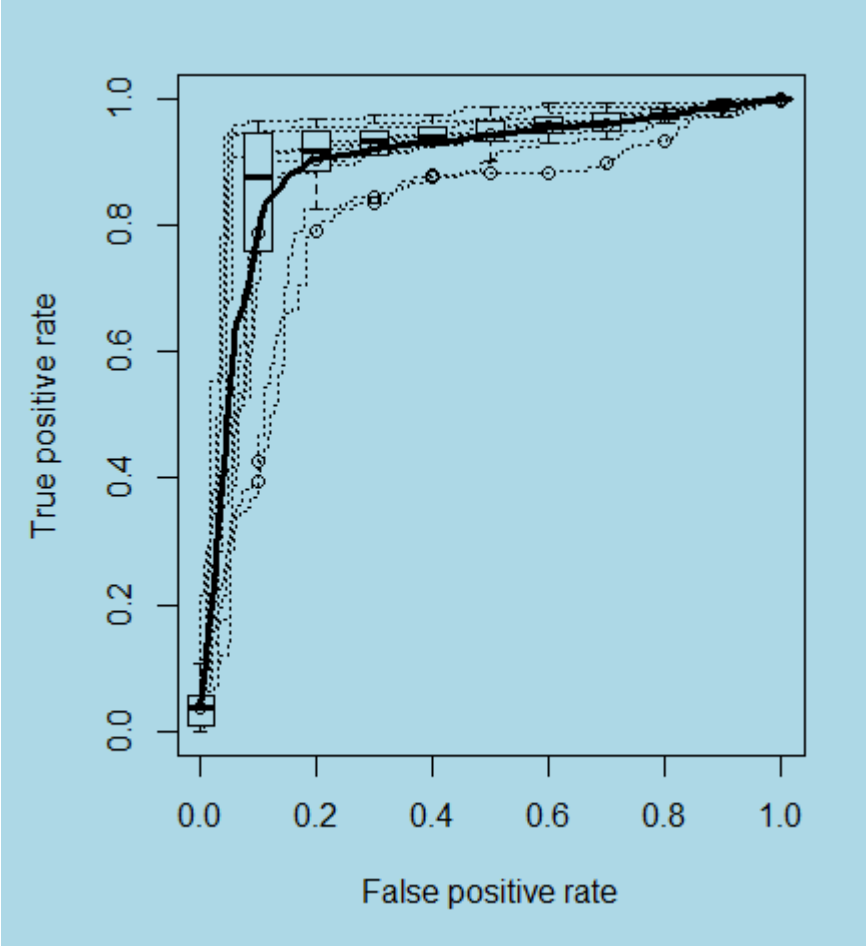
4.3. Пример с несколькими выборками

Покажем различные возможности пакета на примере данных ROCR.xval. Данные получены при 10-fold CV.

```
> data(ROCR.xval)
> pred <- prediction(ROCR.xval$predictions, ROCR.xval$labels)
> perf <- performance(pred,"tpr","fpr")
> plot(perf,col="black",lty=3)
```



```
> plot(perf,lwd=3,avg="vertical",spread.estimate="boxplot",add=TRUE)
```



5. Список литературы

- http://cran.gis-lab.info/web/packages/ROCR/ROCR.pdf
- http://rocr.bioinf.mpi-sb.mpg.de
- http://alexanderdyakonov.narod.ru/upR.pdf